REMARKS

Applicant has requested the cancellation claims 1 and 13 without prejudice. Further, Applicant has presented new claims 15-17, and accordingly, request a new prior art search. Applicant has also submitted amendments to claims 2 and 14. In addition, Applicant submits that the new language in the new claims and in the amended claims do not include new matter.

Canceled Claims

Please cancel claims 1 and 13 without prejudice.

New Claims

Please add new claims 15-17.

Amended Claims

Please amend claims 2 and 14 as indicated above.

Cited References

Broder et al.

Broder et al. is directed to using *load balancing* to distribute workloads over networks of workstations or servers or mirrored sites, (col. 1, lns. 1-17). A client is used as a task directing unit that is interconnected to a plurality of resources, (col. 2, lns. 51-55), where the resources may be processing units that are all part of single multiprocessor computing device such as a high power workstation with multiple processors, or may be each part of a respective single processor computer, which together form a bank or network of servers or workstations, (col. 2, lns. 42-49). The task directing unit is configured to obtain *load information* of each of a number of resources selected uniformly at random, (col. 2, lns. 55-57). The number of randomly selected resources should be substantially less than the total number of resources, (col. 3, lns. 8-10). The task directing unit queries each of the randomly selected resources for *load information* and each of the randomly selected resources respond to the query with its loading information, (col. 2, lns. 58-63). The task directing unit identifies, from among then randomly selected resources, the *least loaded of the resources* selected for processing a task, (col. 3, lns. 28-37), by selecting them at random, and then determining which of such resources is *least loaded.* Broder et al. specifically describes *load information* as task loadings within respective queues, (col. 6, lns. 2-4). Broder et al. further discloses that it is desirable that the queues associated with each of the servers be equally loaded with tasks (col. 4, lns. 32-34). As such, Broder et al. discloses load

balancing based simply on the processing requirements represented by what is currently in a group of particular tasks queues, and does not disclose, at least, "... allocating ... available processing resources among the tasks based on *the capabilities* of each of the available processing resources and the processing requirements of each of the tasks," or "keeping track, remotely from the resources, of the capabilities of all available processing resources; [and] identifying available processing resources in the homogeneous multiprocessor environment based solely on the capabilities kept track of remotely."

Fitch et al.

Fitch et al. is directed to a system for process scheduling from within a current context and switching contexts only when the next scheduled context is different. A control context is a thread, process or task, where a context is in control of a machine environment until it yields, is blocked waiting for something, or is interrupted (descheduling events), (col. 1, lns. 42-49). Fitch et al. discloses a parallel processing environment having multiple address spaces, (col. 3, lns. 53-55), each address space containing application code and kernel code, (Fig. 3), and the context scheduling management system described in Fitch et al. is directed to enhancing the processing within a single address space of the parallel data processing environment, (col. 3, lns. 63-66). Further, an address space can only execute one context (task) at any one time, and if a new context (task) arrives at the address space, the currently running context (task) determines whether the new pending context (task) has priority, and if it does, deschedules the currently running context (task) in favor of the new pending task, (col. 4. lns. 3-34). Further, although Fitch et al. discloses that a control system/kernel understands that it is part of a larger system, i.e., is in a parallel data processing environment, Fitch et al. is does not disclose providing functionality of a first *application-specific subsystem*, (i.e., video processor), to a processor, and that such processor is then associated with a corresponding first input/output device representing the first *application-specific subsystem* (i.e., video device). As such, Fitch et al. does not disclose, teach or suggest, "identifying available processing resources in [a] homogeneous multiprocessor environment ...; allocating the available processing resources among the tasks; providing to the available processing resources *functional programs* and initial data corresponding to the tasks; [and] performing the tasks *using the available processing resources* to produce resulting data."

Frankel et al.

Frankel et al. is directed to a real-time operating system and virtual digital signal processor for the control of a digital signal processor. The system utilizes abstract objects arranged in hierarchical fashion to enable a high level programming language to be used in accessing a wide variety of available functions. (Summary of The Invention). These hierarchical objects are supported by a real-time, multi-tasking system core that manages memory, interrupts, and task-switching. (Id.) The system also includes capabilities of performing I/O operations on an attached host processor file system. (Id). The user of the system is thus not required to be cognizant of the underlying hardware resources, but can program concisely in high level language to carry out operations in ways for which the DSP is best suited. (Id). Further, the system utilizes a kernel providing I/O communications and communication functions, but does not disclose executing such a kernel on multiple processors. Applicants submit that although Frankel et al. discloses a kernel providing I/O communications and communication functions, it does not discuss a plurality of processors executing such kernels, and that Fig. 1 indicates a lack of presence of multiple processors where only a single CPU 11 is shown. At least for such reason, Applicants submit that Frankel et al. does not disclose, teach or suggest Applicants' claimed subject matter.

Hardwick

Hardwick is directed to a dynamic load balancing among processors in a parallel computer. The system includes the determining, at a processor who has just received a task, the computational cost of executing such task, and if it is too expensive to ship it to another processor for execution. (Summary Of The Invention). The system is implemented using a nested parallel programming model called the team parallel model and supports a nested parallel language that can be used to implement irregular device-and-conquer programs on parallel computers.

New Claims 15-17

Independent Claim 15

Applicants respectfully submit that neither Broder et al., Fitch et al. discloses, Frankel et al., nor Hardwick teaches or suggests, whether considered alone or in combination, Applicants' claimed subject matter including, inter alia:

> "...queuing tasks; identifying available processing resources in the homogeneous multiprocessor environment; *allocating* the available processing resources among the tasks *based on the capabilities* of each of the available processing resources and the processing requirements of each of the tasks; providing to the available processing resources functional programs and initial data corresponding to the tasks; performing the tasks using the available processing resources to produce resulting data,"

(Claim 15). Applicants submit that Broder et al. is instead directed to a system which is specifically limited to the *load balancing* of tasks or jobs that are queued up to be serviced by a resource, (col. 2, lns. 65-66). Broder et al. discloses that once queried, a randomly selected resource responds to the contacting task directing unit with only *load information representing its loading*, (col. 2, lns. 51-63). Further, Broder et al. devotes FIGS. 2-6 to system behavior as it relates to load balancing, or more specifically, the time during which a task is queued and serviced for various update intervals of the loading information for each of the resources or servers, (col. 6, ln. 63-col. 7, ln. 3). Applicants submit that Broder et al. lacks any disclosure, teaching or suggestion directed to determining, programming or otherwise identifying any particular resource as having any particular capability, more specifically, Broader et al. lacks disclosure directed to any particular resource having a capability, for example, of either having the functionality of a general-purpose processor or having the functionality of any particular application-specific subsystem processor. Further, Applicants submit that none of the remaining cited references disclose, teach or suggest Applicants' claim 15 subject matter. As such, Applicants submit that neither Broder et al., nor any of the cited references, considered alone or in combination, disclose, teach or suggest Applicants' claim 15 subject matter.

Independent Claim 16

Applicants respectfully submit that neither Broder et al., Fitch et al. discloses, Frankel et al., nor Hardwick discloses, teaches or suggests, whether considered alone or in combination, Applicants' claimed subject matter including, inter alia:

> "... A method for providing multimedia functionality in a homogeneous multiprocessor environment comprising the steps of: queuing tasks; keeping track,

> *remotely from the resources*, of the *capabilities* of all available processing resources; identifying available processing resources in the homogeneous multiprocessor environment based *solely on the capabilities* kept track of remotely; allocating the available processing resources among the tasks; providing to the available processing resources functional programs and initial data corresponding to the tasks; performing the tasks using the available processing resources to produce resulting data,"

(Claim 16). Applicants first direct the Examiner's attention to the arguments above regarding claim 15, and submit that none of the cited references disclose, teach or disclose Applicants' claimed use of, at least, the *capabilities* of each processing resources. Further, Applicants submit that the system disclosed in Broder et al. is specifically limited to a load balancing technique/system which requires a task directing unit to obtain load information *remotely* from the randomly selected resources before it can perform its load balancing function, (col. 2, lns. 51-63). As stated in Broder et al., "[e]ach of the randomly selected resources is configured to respond to the query [from the task directing unit] with load information representing its loading," (col. 2, lns. 60-62). Broder et al. also states that "a client could, in theory, query all servers 50-M for the current load information prior to directing the task to a particular one of the servers 50-M for servicing," (col. 5, lns. 8-12). As such, Applicants submit that Broder et al. discloses a system in which loading information is kept locally to the corresponding resource (remotely to the task directing unit), where the task directing unit must contact a resource and wait a reply before directing a task to a resource, and Applicants submit is wholly unlike Applicants' claim 16 subject matter.

Independent Claim 17

Applicants respectfully submit that neither Broder et al., Fitch et al. discloses, Frankel et al., nor Hardwick discloses, teaches or suggests, whether considered alone or in combination, Applicants' claimed subject matter including, inter alia:

> "... a plurality of processors coupled to a bus; an input/output interface coupled to the bus; a plurality of input/output devices coupled to the input/output interface, the plurality of processors processing program code configured to

> perform a plurality of tasks, the program code comprising: program code configured to cause a first portion of the plurality of processors to interact with a first input/output device of the plurality of input/output devices; program code configured to cause a second portion of the plurality of processors to interact with a second input/output device of the plurality of input/output devices; program code configured to cause a second portion of the plurality of processors to emulate a specific microprocessor instruction set; wherein *the first portion of the plurality of processors provide functionality as found in a first application-specific subsystem* and wherein *the first input/output device is the first application-specific subsystem*; and wherein *the second portion of the plurality of processors provide functionality as found in a second application-specific subsystem* and wherein *the second input/output device is the second application-specific subsystem.*"

(Claim 17). Applicants submit that the system disclosed in Fitch et al. is specifically directed to a system for process scheduling from within a current task and switching tasks only when the next scheduled task is different from the current. Although Fitch et al. describes the use of multiple nodes, where such nodes include a CPU, communication interface, memory and a clocking mechanism, (col. 3, lns. 28-), Fitch et al. does not disclose, teach or suggest, at least Applicants' claimed subject matter, including, inter alia, a

> "... plurality of processors processing program code configured to perform a plurality of tasks, the program code comprising: program code configured to cause a first portion of the plurality of processors to interact with a first input/output device ... wherein the first portion of the plurality of processors provide functionality as found in a first application-specific subsystem and wherein the first input/output device is the first application-specific subsystem."

(Claim 17). Further, Applicants submit that none of the remaining cited references disclose, teach or suggest Applicants' claim 17 subject matter. As such, Applicants submit that neither Fitch et al., nor any of the cited references, considered alone or in combination, disclose, teach or suggest Applicants' claim 17 subject matter.

Rejected Claims 2-3, 5-11 and 14

Dependent Claim 2

Applicants acknowledge the earlier Office Action statement that Broder et al. does not disclose the method of claim 2 wherein a plurality of processors of the homogeneous multiprocessor environment are capable of executing a first instruction of a first instruction set and a second instruction of a second instruction set. Further, Applicants also acknowledge the additional statement that Broder et al.'s use of the term "heterogeneous" was meant to apply to different processor speeds, etc. and not the execution of multiple instruction sets.

Next, the most recent Office Action indicates that Fitch et al. clearly discloses execution of different instruction sets by different processors as allegedly evidenced in Fitch et al. (col. 3, ln. 44-col. 4, ln. 14). Applicants submit that such cited to language is absent the disclosure, teaching or suggestion of Applicants' claim 2 subject matter. Further, although such cited to language does make reference to multiple processing nodes, and that kernels that control such nodes understand they are part of a larger system, i.e., a parallel data processing environment, (col. 3, lns. 44-52), and that the operating systems divide a machine environment into different control tasks over time, (col. 4, lns. 3-5), Applicants respectfully submit that such cited to language is absent any discussion regarding a "... plurality of processors of [a] homogenous multiprocessor environment [that] are capable of executing a first instruction of a first instruction set and a second instruction of a second instruction set." (claim 2). Applicants can find no reference, explicit or implied, which would disclose, teach or suggest the use of multiple instruction sets. Further, Applicants submit that none of the remaining cited references disclose, teach or suggest Applicants' claim 2 subject matter.

To the extent that the Office Action suggests that if one of ordinary skill in the art would combine Border et al. and Fitch et al. to achieve Applicants' claimed subject matter, Applicants respectfully submit that the Office Action uses improper hindsight reasoning by suggesting it would have been obvious to modify Fitch et al. to achieve Applicants' claim 2 subject matter, as the Office Action's arguments are based solely upon the teaching or suggestion within Applicants' own disclosure. Applicants submit that there must be some suggestion or motivation, either in the references themselves, or in the knowledge of generally available to one of ordinary skill in the art, to modify the reference as described. Further, to the extent that the

Office Action relies on a position that modifications of Fitch et al., in view of Broder et al., to meet the claimed invention would have been allegedly well within the ordinary skill of the art at the time Applicants' invention was made, because the references, and/or the knowledge of one skilled in the art, were individually known to those of skilled in the art, is not sufficient to establish a prima facie case of obviousness without some objective reason to combine the teachings of the references. Applicants submit that if one of ordinary skill in the art at the time of Applicants' invention were to read Fitch et al., including that disclosed in Broder et al., such a person would not be in possession of Applicants' claimed subject matter.

Applicants respectfully reassert the arguments made above regarding claim 15. Further, Applicants submit that because claim 2 depends from claim 15, and as a dependent claim therefrom, claim 2 is allowable for at least the reasons claim 15 is allowable. Applicants further submit, argued in part at least immediate above, that claim 2 is also allowable in light of the presence of novel and non-obvious elements contained in claim 2 that are not otherwise present in claim 15.

Dependent Claim 3

Applicants submit that simply because Fitch et al. may allow for multiple contexts (tasks) to run on a particular node, that Fitch et al. does not therefore disclose that a processor of such systems are capable of executing instructions of different instruction sets where a particular instruction from one set share an identical bit pattern of a second instruction set, but perform different operations. A task simply can be considered a particular program or application, and just because such program can be run on any particular processor on any particular node, in no way implies by itself that different instruction sets are available or that any single processor may interpret identical bit-patterned instructions in a different manner depending on any particular circumstance. Applicants submit that simply because Fitch et al. may be modified to allegedly work with different instructions sets, does not by itself render Applicants claimed subject matter obvious, as there must be a suggestion of desirability for the modification, either in the prior art or in the level of skill of one or ordinary skill in the art. As such, Applicants submit that Applicants' claim 3 is allowable as written. Further, Applicants submit that none of the remaining cited references disclose, teach or suggest Applicants' claim 3 subject matter.

Applicants respectfully reassert the arguments made above regarding claim 15. Further, Applicants submit that because claim 3 depends from claim 2, and as a dependent claim therefrom, claim 3 is allowable for at least the reasons claim 2 is allowable. Applicants further submit, argued in part at least immediate above, that claim 3 is also allowable in light of the presence of novel and non-obvious elements contained in claim 3 that are not otherwise present in claim 2.

Dependent Claim 5

Applicants respectfully reassert the arguments made above regarding claim 2, 3 and 15. Further, Applicants wish to re-emphasize their argument that simply accommodating changes between currently executing tasks or contexts does not disclose, teach or suggest a situation where different tasks operate using different instruction sets. Applicants submit that the Office Action makes an unsupported assertion when where it states that "[b]y switching contexts, the program is essentially converted from one instruction set to another." Applicants direct the Examiners attention to the following language from Fitch et al.:

> "A context switch is performed by a conventional 'context switcher' function, which typically entails saving the entire state of a descheduled context, updating kernel scheduling structures, and then restoring an entire state of an incoming control context."

(col. 4, lns. 34-42), and submit that the saving and restoring of states does not in itself suggest the introduction of a different instruction set. More typically, such states represent the status of different variables, tables or other data typically associated with the current status of a running program (or paused program), and in no way represent a switch in instruction sets. Applicants submit that if Fitch et al. contemplated changing instruction sets, it would have been either implied or stated explicitly, and Applicants respectfully submit that neither is present in such reference. As such, Applicants request allowance of claim 5 as currently written.

Applicants also submit that because claim 5 depends from claim 3, and as a dependent claim therefrom, claim 5 is allowable for at least the reasons claim 3 is allowable. Applicants further submit, argued in part at least immediate above, that claim 5 is also allowable in light of the presence of novel and non-obvious elements contained in claim 5 that are not otherwise present in claim 3.

Dependent Claim 6

Applicants respectfully reassert the arguments made above regarding claim 15. Further, Applicants submit that because claim 6 depends from claim 3, and as a dependent claim therefrom, claim 6 is allowable for at least the reasons claim 3 is allowable. Applicants further submit, argued in part at least immediate above, that claim 6 is also allowable in light of the presence of novel and non-obvious elements contained in claim 6 that are not otherwise present in claim 3.

Dependent Claim 7

Applicants respectfully reassert the arguments made above regarding claim 15. Further, Applicants submit that because claim 7 depends from claim 3, and as a dependent claim therefrom, claim 7 is allowable for at least the reasons claim 3 is allowable. Applicants further submit, argued in part at least immediate above, that claim 7 is also allowable in light of the presence of novel and non-obvious elements contained in claim 7 that are not otherwise present in claim 3.

Dependent Claim 8

Applicants respectfully reassert the arguments made above regarding claim 15. Further, Applicants submit that because claim 8 depends from claim 3, and as a dependent claim therefrom, claim 8 is allowable for at least the reasons claim 3 is allowable. Applicants further submit, argued in part at least immediate above, that claim 8 is also allowable in light of the presence of novel and non-obvious elements contained in claim 8 that are not otherwise present in claim 3.

Dependent Claim 9

Applicants respectfully reassert the arguments made above regarding claim 15. Further, Applicants submit that because claim 9 depends from claim 8, and as a dependent claim therefrom, claim 9 is allowable for at least the reasons claim 8 is allowable. Applicants further submit, argued in part at least immediate above, that claim 9 is also allowable in light of the presence of novel and non-obvious elements contained in claim 9 that are not otherwise present in claim 8.

Dependent Claim 10

Applicants respectfully reassert the arguments made above regarding claim 15. Further, Applicants submit that because claim 10 depends from claim 9, and as a dependent claim therefrom, claim 10 is allowable for at least the reasons claim 9 is allowable. Applicants further submit, argued in part at least immediate above, that claim 10 is also allowable in light of the presence of novel and non-obvious elements contained in claim 10 that are not otherwise present in claim 9.

Dependent Claim 11

Applicants respectfully reassert the arguments made above regarding claim 15. Further, Applicants submit that because claim 11 depends from claim 8, and as a dependent claim therefrom, claim 11 is allowable for at least the reasons claim 8 is allowable. Applicants further submit, argued in part at least immediate above, that claim 11 is also allowable in light of the presence of novel and non-obvious elements contained in claim 11 that are not otherwise present in claim 8.

Dependent Claim 14

Applicants respectfully reassert the arguments made above regarding claim 17. Further, Applicants submit that because claim 14 depends from claim 17, and as a dependent claim therefrom, claim 14 is allowable for at least the reasons claim 17 is allowable. Applicants further submit, argued in part at least immediate above, that claim 14 is also allowable in light of the presence of novel and non-obvious elements contained in claim 14 that are not otherwise present in claim 17.

Rejected Claim 4

Dependent Claim 4

Applicants first direct the Examiner's attention to the arguments above regarding claims 2, 3 and 15, and submit that for at least the arguments contained therein, that Applicants' claim 4 subject matter is also allowable as written.

Further, Applicants acknowledge the earlier Office Actions' statement that "neither Broder nor Fitch specifically disclose the method of claim 4 wherein a first processor of the plurality of processors executes an input/output kernel program, the input/output kernel program

including a first portion expressed using the first instruction set and a second portion expressed using the second instruction set."

Although Frankel et al. discloses a kernel providing I/O communications and communication functions, it does not discuss a plurality of processors executing such kernels. Further, Fig. 1 indicates a lack of presence of multiple processors where only a single CPU 11 is shown. Further, as discussed above regarding claim 2, neither Broder et al., nor Fitch et al. disclose, teach or suggest the use of different instruction sets by different processors. Applicants submit that the Office Action is attempting to use impermissible hindsight based on Applicants' disclosure to indicate the obviousness of Applicants claim 4. Further, Applicants submit that simply because Frankel et al. may be modified to allegedly work with different instructions sets, does not by itself render Applicants claimed subject matter obvious, as there must be a suggestion of desirability for the modification, either in the prior art or in the level of skill of one or ordinary skill in the art.

Applicants submit that at least because claim 4 depends from claim 3, and as a dependent claim therefrom, claim 4 is allowable for the reasons claim 3 is allowable. Applicants further submit that claim 4 is also allowable in light of the presence of novel and non-obvious elements contained in claim 4 that are not otherwise present in claim 3.

Rejected Claim 12

Dependent Claim 12

Applicants first direct the Examiner's attention to the arguments above regarding claims 2, 3 and 15, and submit that for at least the arguments contained therein, that Applicants' claim 12 subject matter is also allowable as written.

Further, Applicants acknowledge the earlier Office Actions' statement that "neither Broder nor Fitch specifically disclose the method of claim 4 wherein a first processor of the plurality of processors executes an input/output kernel program, the input/output kernel program including a first portion expressed using the first instruction set and a second portion expressed using the second instruction set."

Further, as discussed above regarding claim 2, neither Broder et al., nor Fitch et al. disclose, teach or suggest the use of different instruction sets by different processors. Here, where the Office Action is attempting to combine the teachings of Broder et al, Fitch et al. and

Hardwick, Applicants submit that the Office Action is attempting to use impermissible hindsight based on Applicants' disclosure to indicate the obviousness of Applicants claim 12. Further, Applicants submit that simply because Hardwick may be modified to allegedly work with different instructions sets, and may also be modified dynamically adjusting processing resources utilizing different instruction sets, does not by itself render Applicants claimed subject matter obvious, as there must be a suggestion of desirability for the modification, either in the prior art or in the level of skill of one or ordinary skill in the art.

Applicants also submit that at least because claim 12 depends from claim 3, and as a dependent claim therefrom, claim 12 is allowable for the reasons claim 3 is allowable. Applicants further submit that claim 12 is also allowable in light of the presence of novel and non-obvious elements contained in claim 12 that are not otherwise present in claim 3.

CONCLUSION

For the foregoing reasons, withdrawal of the rejections and allowance of the claims is respectfully requested. If there are any questions or comments regarding this response, the Examiner is encouraged to contact the undersigned at 312-609-7500.

Respectfully submitted,

By: _____
Brent A. Boyd
Reg. No. 51,020

Dated: December 8, 2003

Vedder, Price, Kaufman & Kammholz, P.C.
222 North LaSalle Street
Chicago, Illinois 60601
Telephone: (312) 609-7500
Facsimile: (312) 609-5005